White Paper Report

Report ID: 107198

Application Number: HK-50032-12

Project Director: David Miller (dmill1951@gmail.com)

Institution: University of South Carolina Research Foundation

Reporting Period: 9/1/2012-8/31/2015

Report Due: 11/30/2015

Date Submitted: 12/1/2015

November 30, 20125
Travis Mullen, with David Lee Miller and Song Wang

The PARAGON Project White Paper

PARAGON (tundra.csd.sc.edu/PARAGON) is a software system capable of intelligent collation and difference detection among materials from multiple repositories, digitized according to varying standards with a range of methods and equipment. The Center for Digital Humanities and the Computer Vision Lab at the University of South Carolina collaborated on the project, funded by an NEH Digital Humanities Implementation Grant (HK-50032-12). Source code for the software is available for download at https://github.com/CDH-SC/paragon.

Personnel

Project Directors: David Lee Miller, SongWang
Project Manager: Travis Mullen
Algorithm Developers: Dhaval Salvi, Kang Zheng, Lingxi Zhou
User Interface Developer: Eric Gonzalez
Front-end Programmer: Collin Haines
User Interface Designer: Carly Keith

Goals

As historical documents are digitized in greater numbers, humanists and computational scientists are challenged to develop tools and procedures that will take advantage of this new form of access to archival materials. Much has been done to study the process of digitization, and with it the problems of data loss and preservation; computational methods to preserve and study digital assets held by museums and libraries have made significant headway in identifying best practices for digital collection and curation. Nevertheless, the methods and equipment used to digitize materials still vary widely, as do the policies and standards of various repositories. Such variety in the conditions of collection and in the resultant digital images presents major challenges to the comparison and analysis of materials from multiple repositories. PARAGON (from the early modern Italian *paragone*, a "comparative analysis between alternatives resulting in a choice"[1]) addresses these challenges with a method of intelligent digital collation and difference detection.

Collation of early modern printed materials is crucial to the disciplines that study them because manuscript copy seldom survives. Print typically offers the only access to what an author wrote —yet extant copies differ in many places, while in others they are manifestly erroneous. Early editors combined erudition with sometimes-inspired guesswork in emending texts, but they did so haphazardly; systematic collation began at the turn of the twentieth century with the rise of the "New Bibliography." Modern editorial theory, based on a rigorous empirical method informed by careful study of the practices and conditions in print shops, developed out of the New Bibliography when D. F. McKenzie demonstrated that seemingly innocuous commonsense assumptions woven through the deliberations of earlier bibliographers had in fact given rise to fantastical creatures—"Printers of the Mind," as the title of one seminal essay put it.

---

[1] *Oxford English Dictionary*, s.v. "paragon ."

[2] Steven Escar Smith, "Armadillos of Invention: A Census of Mechanical Collators." *Studies in Bibliography* 55 (2002): 133-171. (http://etext.virginia.edu/bsuva/sb/)

Textual collation provides one kind of evidence indispensable to modern bibliographic method: the systematic comparison of multiple copies from a single print run in order to specify differences among them. This evidence is indispensable because proofreading in early modern print shops was done on the fly, with corrections made by stopping the presses to remove and unlock the chase containing blocks of type, a practice that occasionally introduced new errors in the process of correcting old ones. Since paper was too expensive to waste, sheets representing different states of the text were gathered and bound together at random. For any given publication surviving from this period, it is therefore possible that no two copies from the same print run will be identical. Collation isolates differences so that the bibliographer can use them to build a detailed genealogy of the text's production. It is on the basis of such a genealogy that editorial decisions are made.

Textual collation is an indispensable but labor-intensive step in the study of print materials. Textual scholars and bibliographers have developed a number of ingenious tools to aid in this process, but it remains both expensive and time-consuming, involving travel to distant repositories for the painstaking visual examination of multiple original copies.[2] Previous efforts to computerize collation depended on first transcribing the texts to be compared, introducing into the process a layer not only of labor and expense but also of potential error. PARAGON instead automates the first stages collation directly from scanned images of the original text, dramatically speeding the process; it also affords new functionalities for coping with variations in the quality and rendering of digital materials captured in different ways at different times.

History

PARAGON builds upon "The Sapheos Project: Transparency in Multi-image Collation, Analysis, and Representation," a successful prototype funded by the National Endowment for the Humanities in 2009 (HD-50880-09) and developed by the Center for Digital Humanities and the Computer Vision Lab at the University of South Carolina (Project Directors Song Wang and Randall Cream).[3] Sapheos demonstrated that SIFT (scale-invariant feature transform) and TPS (Thin-Plate-Spline) algorithms can be combined to automate the collation of images whether scanned, camera-taken from different heights or angles, rotated, differently lighted, or even slightly warped.

Methods

Collation Algorithm

The collation algorithm—which is distinct from the preprocessing and dewarping algorithms—has three main parts: fitting the template and detecting its main feature points, fitting the target image to the template (mostly an adjustment of scale) and calculating its main feature points, and finally comparing the two images pixel by pixel using the feature points as references.[4] The

---

[2] Steven Escar Smith, "Armadillos of Invention: A Census of Mechanical Collators." *Studies in Bibliography* 55 (2002): 133-171. (http://etext.virginia.edu/bsuva/sb/)

[3] The Sapheos white paper is available for download at https://securegrants.neh.gov/publicquery/main.aspx?f=1&gn=HD-50880-09.

[4] The algorithm is written in C and integrated into our Django/Python based interface by being compiled into Shared Object Libraries.

comparison algorithm works by conducting a pixel-by-pixel comparison of the two images with the pre-determined feature points serving as references to help reduce the number of false positives.

In the first phase, PARAGON adjusts the template image to better match the target(s) so that comparisons will be more accurate. This process may involve trimming the edges or reducing the size of the template (when one image is larger than the other, the algorithm's preference is to reduce the size of one image in order to avoid distortion). After the dimensions of the image are more closely aligned, the algorithm determines points in each image to compare against points in the other image. The images then move into the second phase of the collating process, transformation.

In phase two, PARAGON creates a new version of the target image, adjusting it on a pixel level to conform more closely to the template image based on the feature points (those that were calculated in the first step, and new points determined during this phase). This transformation enables more accurate comparison in the third phase of the collating process.

During the third phase, collation, the template is compared to the transformed image in several ways. The algorithm does a pixel-to-pixel comparison guided by the feature points established in the first two phases. The differences discovered by this comparison are then sorted based on "unit" recognition (for example, that of a letter or word); those that do not represent units (i.e. specks of dust) are removed from the results. The results of collation are sets of coordinates that draw boxes on the screen over the transformed image where PARAGON has detected differences between the template and target images:

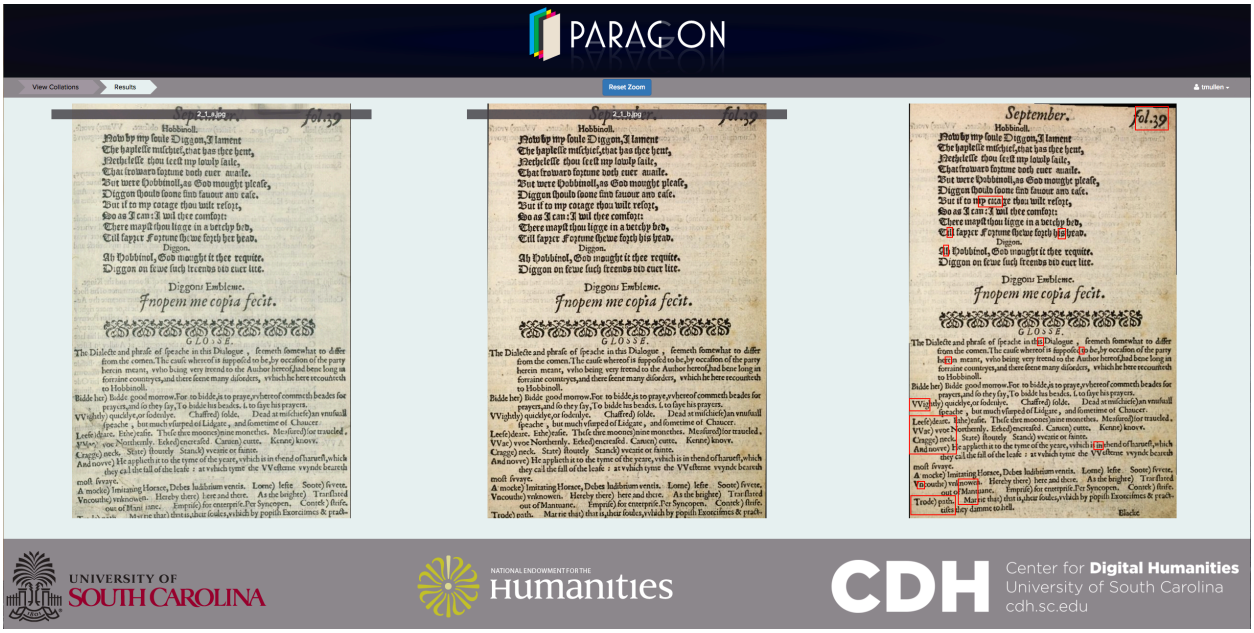

Figure 1. Results of collation are displayed in boxes drawn over the page image.

The user interface offers three options for collating images of text. Single collation works on isolated pairs of page-images; multi-collation compares a single "template" page-image to

multiple images of the same page taken from different copies. There is also a book-collation option that compares pages on a one-to-one basis for different copies of the same book:

Figure 2. PARAGON offers single-pair collation, multiple-image collation, and book collation.

Instructions for implementing these options are available in the User Guide offered on the project web site.

The algorithm has undergone many changes based on user feedback and testing. Many of the revisions are targeted at reducing what we call "false positives" that tend to isolate differences inherent in images and in the objects themselves (paper quality and damage, dirt on the pages, ink bleed through, differences in inking, lighting differences, etc.). The difficulty here is in producing an algorithm that is sensitive enough to encompass all significant differences, especially in small cases like punctuation, but not so sensitive that it determines all differences to be significant. The current revision of the algorithm is precise enough to detect differences on even the level of punctuation while ignoring a good bit of noise, but further refinement is needed. During collation, all processes are passed to a background processor called Celery that allows for thousands of simultaneous collations and greatly improves the efficiency and speed of collation. This adjustment also allows the user to begin a collation and have it finish without having to stay logged in or on the website at all.

Dewarping Algorithm

Repositories holding rare books are often reluctant to flatten the pages during digitization, since to do so risks damage to the binding. The resulting page curvature in digital scans presents special challenges to automating collation.

Our developers responded to this challenge with an algorithm for dewarping page images.[5] This algorithm works by detecting lines of text, measuring the length of the lines, determining their center, and calculating the arc. The calculation is then used to adjust the text back toward a

[5] This algorithm is also written in C and compiled into SO Libraries.

straight line. Like the collation algorithms, this one runs in the background. It automatically determined whether dewarping is needed and skips the image if it is not.[6]

The dewarping process works well enough that it has been shared to the Washington University Lab for use with images in the Spenser Archive.

Preprocessing

In the effort to reduce "noise" and the false positives that it causes, the development team added a step prior to dewarping and collation. This preprocessing procedure converts scans to black and white, crops them, removes extraneous data (e.g., the edges of subsequent pages), and reduces the amount of bleed through.

Future Directions

The PARAGON development team has achieved most of the objectives stated in our 2012 grant proposal, but we hope to make further improvements in performance and ease of use.
Most important is to continue fine-tuning the collation algorithm to reduce false positives. We think this can be done by limiting the size of the units to be compared: instead of page-to-page, we should be able to isolate stanzas, paragraphs, or individual lines—perhaps even individual words—to enable the algorithm to distinguish between printed characters and "noise" (dirt, damage to the page, bleed through, etc.). The process will still involve pixel-to-pixel comparison rather than character-recognition, but we hope that bounding boxes can be used to improve results by limiting the surface areas compared.

Our next priority is to address the challenge of comparing visual images. Because the dewarping algorithm uses lines and white space to calculate the arc of curvature, it works only on printed text. Two projects on the horizon offer opportunities for extending the software's current range of application: a proposal for comparing early modern map-images to track the changes that appear as maps are created from other maps, and another to compare the patterns on pottery shards from early Native American archaeology sites in Georgia and South Carolina.
On a smaller scale, we need to automate the export of text collation results. At present, results are given as a set of colored boxes superimposed on the image, but these boxes are drawn in the browser and not directly on the image. We also need to increase the size of zip and pdf files our users can upload for book-to-book collation.

These and other improvements can be implemented as we test the software on a wider range of corpora, including large-scale comparisons. We continue to receive inquiries from prospective users with various datasets (e.g., a Portuguese epic poem or the works of Jonathan Swift).
We do intend to seek support for further development of the software.

[6] The process will skip images that do not have enough text for the algorithm to work on, a limitation that needs to be addressed in future development of the software.

Bibliography

Agata, Mari, 2005. “Toward Collation with Digital Images.” *Library and Information Science* No. 53.

Bookstein, F. L. 1989. “Principal warps: Thin-plate splines and the decomposition of deformations.” *TPAMI* 11.6: 567–585.

Coustaty, M., J.-M. Ogier, R. Pareti, and N. Vincent, 2009. “Drop caps decomposition for indexing a new letter extraction method.” *ICDAR* 476–480.

Liang, J., D. DeMenthon, and D. Doermann, 2006. “Camera-based document image mosaicing.” *ICPR*, 476–479.

Lowe, D. G., 2004. “Distinctive Image Features from Scale-Invariant Keypoints.” *International Journal of Computer Vision*, 60: 91-110.

Marinai, S., E. Marino, and G. Soda, 2007. “Exploring digital libraries with document image retrieval.” *ECDL*, ser. Lecture Notes in Computer Science, ed. L. Kov´acs, N. Fuhr, and C. Meghini, vol. 4675, 368–379. Springer.

McKenzie, D. F, 1969. “Printers of the Mind: Some Notes on Bibliographical Theories and Printing-House Practices.” *Studies in Bibliography* 22: 1-75. (http://www.jstor.org/stable/40371475)

Miller, David Lee, 2007. “Building a Spenser Archive—One Scan at a Time.” *Duke University Libraries* 20:2/3: 14-19.

Randen, Trygve, and John Hakon Husoy, 1994. “Segmentation of text/image documents using texture approaches.” *Proceedings of Nobim-konferansen*, 60-67.

Roy, P.P., Pal, U., Llados, J. and Delalandre, M., 2009. “Multi-Oriented and Multi-Sized Touching Character Segmentation Using Dynamic Programming.” *ICDAR*

Raabe, Wesley. 2008. “Collation in Scholarly Editing: An Introduction.” (http://wraabe.wordpress.com/2008/07/26/collation-in-scholarly-editing-an-introduction-draft/)

Smith, Steven Escar 2002. “Armadillos of Invention: A Census of Mechanical Collators.” *Studies in Bibliography* 55: 133-171. (http://etext.virginia.edu/bsuva/sb/)

Van Beusekom, J., F. Shafait, and T. M. Breuel, 2007. “Image matching for revision detection in printed historical documents.” *DAGM*, 507–516.